Bolt Beranek and Newman Inc.

AD-A132 690

Report No. 5408

Combined Quarterly Technical Report No. 30

Pluribus Satellite IMP Development
Mobile Access Terminal Network

DTIC
SELECTE
SEP 2 1 1983
S D
A

August 1983

Prepared for:
Defense Advanced Research Projects Agency

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A132 98	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Combined Quarterly Technical Report No. 30		5. TYPE OF REPORT & PERIOD COVERED Quarterly Technical 6/1/83 to 8/31/83
		6. PERFORMING ORG. REPORT NUMBER 5408
7. AUTHOR(s) S. Blumenthal, D. McNeill		8. CONTRACT OR GRANT NUMBER(s) MDA903-80-C-0353 N00039-81-C-0408
9. PERFORMING ORGANIZATION NAME AND ADDRESS Bolt Beranek and Newman Inc. 10 Moulton Street Cambridge, MA 02238		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Arpa Order No. 3214
11. CONTROLLING OFFICE NAME AND ADDRESS Defense Advanced Research Projects Agency 1400 Wilson Boulevard Arlington, VA 22209		12. REPORT DATE September 1983
		13. NUMBER OF PAGES 21
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) DSSW Room ID The Pentagon Washington, DC 20310 — NAVALEX Washington, DC 20360		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

APPROVED FOR PUBLIC RELEASE/DISTRIBUTION UNLIMITED

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Computer networks, packets, packet broadcast, satellite communication, gateways, UNIX, Pluribus Satellite IMP, shipboard communications, ARPANET, Internet.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This Quarterly Technical Report describes work on the development of Pluribus Satellite IMPs and on shipboard satellite communications.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Report No. 5408

COMBINED QUARTERLY TECHNICAL REPORT NO. 30

PLURIBUS SATELLITE IMP DEVELOPMENT
MOBILE ACCESS TERMINAL NETWORK

August 1983

This research was supported by the Defense Advanced Research Projects Agency under the following contracts:

MDA903-80-C-0353, ARPA Order No. 3214
N00039-81-C-0408

Submitted to:

Director
Defense Advanced Research Projects Agency
1400 Wilson Boulevard
Arlington, VA 22209

Attention: Program Management

Table of Contents

1 INTRODUCTION

This Quarterly Technical Report is the current edition in a series of reports which describe the work being performed at BBN in fulfillment of several ARPA work statements. This QTR covers work on several ARPA-sponsored projects including (1) development of the Pluribus Satellite IMP; and (2) development of the Mobile Access Terminal Network. This work is described in this single Quarterly Technical Report with the permission of the Defense Advanced Research Projects Agency. Some of this work is a continuation of efforts previously reported on under contracts DAHC15-69-C-0179, F08606-73-C-0027, F08606-75-C-0032, MDA903-76-C-0214, MDA903-76-C-0252, N00039-79-C-0386, and N00039-78-C-0405.

2 PLURIBUS SATELLITE IMP DEVELOPMENT

2.1 Introduction

BBN efforts during the quarter concentrated on Wideband Network operations, participation in the Wideband Network task force activities, support of the SRI/NTA Internet speech exercise, and BSAT software development.

2.2 Wideband Network Task Force Activities

At the March 1983 Wideband Network meeting, a task force with representatives from Lincoln Laboratory, ISI, Linkabit, and BBN was created to investigate the remaining systems level problems that have impacted network performance. The goal of the task force is to achieve reliable network operation at 3.088Mb/s. The task force convened for the first time at Lincoln during the week of April 25. During that visit, it was found that the ESI was not properly processing aggregated multi-rate bursts. Linkabit was able to fix this problem in their laboratory, and they installed this fix in the ISI ESI in early May. The task force next convened at ISI during the week of May 16. At that time, it was verified that the ESI was aggregating multi-rate bursts properly. During this task force visit, an ESI spurious uplink interrupt problem was uncovered. Linkabit took the ISI

ESI back to their laboratory to work further on this problem. The problem was fixed and the ESI was returned to ISI on June 3.

During the May 16 task force visit to ISI, a bandpass filter was inserted into the downlink path between the low noise amplifier (LNA) and the downconverter. The filter effectively eliminated the aircraft radar altimeter RFI intermodulation noise that had been identified by Probe Systems' analysis. Western Union plans to procure and install these filters at all sites.

The task force convened for the third time at ISI during the week of June 27. A bug was found in the PSAT datagram fragmentation software. Packets were occasionally being fragmented in the middle of the CRC words, where no fragmentation is allowed. The problem was traced to an incorrect constant, and this was fixed by BBN during the task force visit. A second PSAT bug was found that could not be fixed during the task force visit. Occasionally the PSAT would pass the ESI a control packet with the symbol rate set at 193 ksymbols and the burst length equal to zero. Back at BBN it was found that this bug was caused by several routines accessing the same buffer without proper locking taking place. It was determined that this improper buffer accessing by two routines was also causing datagram packets to be generated with occasional checksum errors. It turned out that word 16 in a buffer could contain the channel

symbol rate and burst length in an ESI control packet and the packet checksum in a datagram packet header.

Western Union worked with Lincoln during the June 27 task force visit to ISI to adjust the antenna subreflector. Following the subreflector adjustment, the site realized a round trip signal gain of 2 db.

There were no task force visits during July. Task force activity concentrated on two-site testing at 1.5 Msymbols, with BPSK and QPSK modulation, and with mixed rate coding in preparation for reporting to DARPA and DCA on August 1.

2.3 Wideband Network Operations, Maintenance and Status

Several earth station and ESI problems were encountered during the quarter. The DCEC ESI was returned to Linkabit for repairs during the early part of May and remained there throughout the month. The Lincoln ESI developed an intermittent hardware problem on June 1. Once the DCEC unit had been repaired and upgraded during the early part of June, it was shipped to Lincoln to serve as a backup for the SRI/NTA Internet speech exercise. On June 8, the ISI earth station's upconverter failed. It was repaired by Western Union on June 9.

The SRI ESI was not upgraded during the quarter to work reliably at 1.5 Msymbols and thus the site had to be looped off the channel for most of the quarter to allow Lincoln and ISI to test at the higher rates. On several occasions throughout the quarter, however, the channel was operated at 772 Ksymbols to allow SRI and Lincoln to prepare for and conduct the SRI/NTA Internet speech exercise.

There were several minor PSAT hardware problems during the quarter. On May 9, one of the ISI Super SUE pollers failed. The software was patched to use the spare until a replacement could be installed. The broken SSP was replaced by BBN during the May 16 task force visit to ISI. One of the processors failed in the DCEC PSAT, and it was replaced by BBN field service on June 3. A satellite modem interface (SMI) failed in the RADC PSAT on June 15. The RADC PSAT's code was patched to use the spare SMI. The RADC PSAT developed a memory problem on July 4. On July 7, BBN replaced a bad memory board and also replaced the broken SMI. On July 11, the ISI PSAT developed a memory problem, and a bad memory board was replaced that afternoon by BBN field service. Also on July 11, the Lincoln PSAT developed an intermittent hardware problem. The problem was finally tracked down to a bad memory ard, which was fixed on July 22.

Throughout the quarter BBN continued testing a new version of the PSAT program. The new version converts the spare code pages into system buffers, increasing the number of system buffers from 240 to 440. This will significantly increase the amount of host datagram traffic that the PSAT can handle. The new version also contains code to collect and process ESI T&M data. Several bugs were found in the new code, from testing by BBN and as a result of the task force activities. Among these bugs were the one in fragmentation routine and the overwriting of the ESI control packet channel symbol and burst length word, which were found as a result of the June 27 task force meeting. A multi-site stream synchronization bug was found and was being worked on by BBN at the end of July.

An acceptance test was successfully completed for the RCA Integrated Node Host at RADC on July 20. The Integrated Node simulates a three-node circuit switch telephone network, with the Wideband Network (via the RADC PSAT) serving as a redundant packet switched link between two of the simulated nodes.

During the week of June 20, a PSAT and a Lincoln Packet-to-Circuit (PCI) interface were successfully installed at Ft. Huachuca, AZ. At the end of the quarter, Western Union was still in the process of installing the earth station, and Linkabit had not yet delivered the ESI.

2.4 BSAT Progress

Significant progress was made in BSAT software development during the quarter. Several major pieces of software that had been under development are now working, and the channel scheduler software was rewritten to move the datagram aggregation function into a separate process.

The External Host module, which includes processes to handle the Host Access Protocol (HAP) and a process to handle the Host synchronous I/O interfaces, was debugged during May. By mid-May, we were able for the first time to connect a Voice Funnel to the BSAT and have its HAP link come up. This indicated that the HostIn, HostOut, HAPControl, and HostIO processes were initially working.

Similar progress was made debugging the Channel Protocol Module (CPM). Writing of initial sections of the uplink and downlink software was completed during the quarter. By late May, we were able to send ranging and leader packets via the uplink and downlink code with the CPM I/O hardware interfaces running internally crosspatched. This path simulated the 1/4 second delay that would be present if the messages were sent via an ESI and the satellite. The CPM scheduler successfully completed initial ranging and declared itself leader. This meant that the Channel Scheduler, CPM transmit (uplink), CPM loopback, and CPM

receive (downlink) processes had begun to work.

With the initial success of the CPM module, we reworked the channel scheduler to separate out the datagram aggregation code. By the end of May, we had succeeded in getting enough of the code to work that we could send messages from the Message Generator via the CPM with the hardware internally crosspatched, through the Echo Host, to the Message Sink. Bugs remained, and the datagram reservation code was not yet being used, but the main path was working.

Improvements were made to the message statistics gathering and printout routines. We now can see the number of messages going to and from external hosts, the number of messag going to and from the channel, whether or not we ever run out of host buffers or CPM buffers, and whether or not various internal queues filling up cause messages to be lost.

In June, we began testing mixed-rate datagram bursts. That code is now working to the extent that the datagram aggregator properly computes the burst size, the scheduler properly schedules it, the uplink correctly performs burst rearrangement of the aggragated messages, and the downlink will divide up the burst back into the component messages.

In the second half of the quarter, we began designing the stream scheduler and stream aggregator processes and debugging the datagram reservation code. We also redesigned the BSAT buffer ownership rules to simplify the code and to provide a means for the HDLC protocol process (now in the design stage) to resend messages efficiently. On the basis of the decision to begin testing the BSAT in the Wideband Network using the PSAT as translator to connect to the ESI-A, we began augmenting the BSAT CPM software to be able to deal with PSAT-format channel bursts.

2.5 BSAT CPM Scheduler Redesign

2.5.1 Motivation

The BSAT is being designed as a high-throughput packet switch. It uses the Butterfly multiprocessor, whose architecture allows adding additional processors to get proportionally higher throughput. To take advantage of additional hardware when it is available, the software must be capable of creating new processes to run on the additional processors.

For an application such as the BSAT, this means that processes should be designed so that a large number of copies of the same basic process can all be run at the same time on separate processors. If a single process is capable of

processing "M" messages per second, creating "P" processes should make the system able to handle as close to M * P messages per second as possible. If this were true of all processes in the system, then, to first order, the system throughput could be made as large as desired simply by adding more processors and configuring the software to use them.

Unfortunately for this goal, the BSAT and PSAT contain a few places where duplication seems to be difficult or impossible. These generally arise from a requirement that something be "scheduled"; i.e , that all events related to that device be strictly ordered according to some parameter. Usually, this means that a single process must be chosen to do the ordering, since proper ordering requires information about all the things to be ordered. Throughput will then be limited by the speed of the scheduling process.

If a scheduling process must exist, then the highest throughput will result if that process is relieved of every computation not directly concerned with the scheduling task. This design rule underlies our changes to the CPM Datagram Scheduler and will also be a central part of our design of the CPM Stream Scheduler and of the Stream Aggregator.

In simplest terms, the CPM Datagram Scheduler has been pared down to contain only those procedures involved with scheduling a

burst on the channel, and giving the burst descriptor to the appropriate ESI uplink process. (More than one process will exist if the data rate of the satellite channel exceeds the rate of a single Butterfly I/O channel.) The Datagram Aggregator contains all the code to collect host messages, compute their size in bits and channel symbols, and aggregate them together into bursts.

The next section describes the new Datagram Scheduler and Datagram Aggregator processes in greater detail. Note that fragmentation has not yet been implemented. Some of the design described below will change when fragmentation is added.

2.5.2 The Datagram Aggregator

The Datagram Aggregator is responsible for aggregating datagrams into bursts, creating the datagram burst reservation block, and computing the burst length and state lengths. It is the first step in the process of sending datagrams over the satellite channel.

The Datagram Aggregator starts to aggregate messages when they appear on the queues from the HPM. Before it puts the first message into the burst, it sets a timer for about 20 milliseconds in the future. Once it has started to fill the

burst, the datagram aggregator continues to aggregate messages into that burst until either it has filled the burst or the timer goes off. Once either of these conditions has been fulfilled, the reservation block is sent to the Scheduler to be scheduled for transmission. For moderate traffic levels, the datagram aggregator should create a new reservation about once per frame.

The current version of the BSAT has a limit on the number of bursts that have been composed and are waiting for their reservations to be transmitted. This limit, RESVTHRESH, is currently set to two. The datagram aggregator attempts to aggregate as many bursts as it can without exceeding RESVTHRESH.

The aggregation procedure begins when there is a message to aggregate and there are less than RESVTHRESH reservations to be transmitted. The process will be awakened, when needed, by a message appearing on the CPM datagram input queues or by the Scheduler after sending a reservation.

The first step in aggregation is to get a reservation block and initialize it. The second step is to set the timer for 20 milliseconds in the future, which is the latest time the burst can be passed to the scheduler. If the last message dequeued did not fit in the previous burst, then it will be the first message considered for aggregation in the current burst.

The reservation block is used to keep track of the burst length, the statewords, and the messages in the burst itself. The burst length is initialized to the length of the header, and the statewords are initially set to zero. As the BSAT adds messages to the burst, it updates the burst length (kept in channel symbols), and the length portion of the statewords (kept in 16-bit words). Each message is appended to the end of the message chain for the reliability at which the datagram is to be sent.

If the datagram aggregator runs out of messages to aggregate without filling the burst, it waits until either another message appears on the CPM datagram input queues or the timer fires. If another message appears, it puts the new messages into the burst reservation. If the timer has fired, the aggregation loop is exited so the reservation can be passed to the Datagram Scheduler.

The two conditions that cause the aggregator to terminate a burst are if the timer fires, or if the addition of the next message would cause the length to exceed the maximum burst length of 16K channel symbols. In either case, the timer is cleared. If the next burst would exceed the maximum burst length, the message is checked to see if it would have been the only message in the burst. If so, the message and the

empty reservation block are discarded, as the message is obviously too long to fit in any burst. Otherwise, the pointer to the message that didn't fit is saved, so that the message can be put in the next burst. The reservation block with its queues of messages to be put in the burst is then passed to the Scheduler process.

It is possible in the future that special considerations could cause messages to be sent in the next burst even if space is left in the current one. An example of a message type that may need to be sent in a separate burst would be a message containing information to be loaded into another site.

2.5.3 The Datagram Scheduler

The datagram scheduler is responsible for scheduling bursts on the satellite channel based on the reservations it has received and timed out. Each reservation to be scheduled is represented by a reservation block on the queue of datagrams whose reservations have timed out and are ready to send. The BSAT will have the original reservation blocks for its own reservations, and dummy reservation blocks for the reservations heard from other sites.

The datagram subframe starts at the end of the setup subframe, and can continue until the beginning of the (minimum) control subframe. The scheduler dequeues the reservations in sequence, testing each in turn to see if it fits in the datagram subframe. This testing and scheduling also includes a certain amount of interburst padding to account for ESI uplink and downlink burst processing and to allow for small site-to-site global time inconsistencies. If a burst does not fit in the datagram subframe, it is returned to the top of the queue, and the datagram scheduling procedure returns.

Once the datagram scheduler has determined that the next burst will fit, it records the time for the ESI to start transmitting the burst. The scheduler also keeps a cumulative record of the amount of the datagram subframe that has been scheduled.

If the scheduled burst is to be transmitted by this BSAT, it gets a burst descriptor and fills in the burst header. Reservation words for other messages may be inserted in the header, and the ranging bit may be set if it is time for this BSAT to range. The pointer to the message chain is set, and the burst descriptor is passed to the CPM transmit process.

Once per frame, the Scheduler will timeout datagram reservations. It must do so before scheduling datagram bursts,

because it must determine which bursts can be scheduled in this frame.

When a reservation is received in a burst, the CPM_Rcve process extracts the necessary reservation information from the burst and stores it in a reservation block. This reservation block is then passed to the Datagram Scheduler. TimeoutDatagramResvs takes the reservations off the queue as they time out. If the timed-out reservation is not from this site, the reservation block is merely passed on to the scheduling queue. If it is from us, the reservation block must be matched up to the original reservation block, as that reservation block contains the queues of messages to be sent in the burst.

The datagram reservation matching process is complicated by the fact that reservation information may be lost when it is transmitted over the satellite channel. There are three cases that may occur. The first is that the site received a reservation that it claims it sent, but it has no record of the reservation. In this case, it is expected that all the other sites received the same reservation, so the burst must be scheduled anyway to maintain channel synchronization. The second case is when the site was expecting to receive another reservation from itself before this one. In this case, the missed reservation must be resent,

and the whole reservation transmission is repeated. This can happen a maximum of MAXSENDS (10) times before the reservation and the messages are discarded. The third case is when the next expected reservation matches the reservation received. In this case, the received reservation is discarded, and the matching reservation block, which has the chains of messages for the burst, is put on the queue for scheduling.

Note that the scheduler assumes reservations are received in the same order at all sites. This requirement is met because only one copy of the datagram reservation is sent at any time, and because all sites share the same downlink.

3 MOBILE ACCESS TERMINAL NETWORK

As part of our participation in the development of the Mobile Access Terminal (MAT) and the MAT Satellite Network (MATNET), we provided support for the system integration, which included the SHIP3 MAT installation aboard the CVN70, and for the ongoing large scale system testing. Other participants in the above activities included E-Systems, ECI Division, in St. Petersburg, Florida; Tracor, Inc., in San Diego, California; and the Advanced Command and Control Architectural Testbed (ACCAT) at the Naval Ocean Systems Center (NOSC) in San Diego, California. In the MATNET project, the Terminal Input Unit (TIU) hardware, the COMSEC equipment, the Black processors, and the radio equipment are ECI's responsibility, while the C/30 Satellite IMPs, the gateway, and the TIU software are BBN's responsibility. Late in the last quarter, our participation in the development of MATNET was reduced to a low-level support while waiting for contract renewal.

When the captain of the carrier USS Carl Vinson (CVN70) requested that NAVELEX restore to operational status the SHIP3 MAT station, which had been hurriedly installed aboard ship before it left CONUS (see BBN Combined Quarterly Technical Report No. 28), representatives from BBN, ECI, Tracor, and NAVELEX were dispatched to the ship on location in the Indian Ocean. Our

tests reconfirmed that the C/30 Satellite IMP was functioning properly; site problems were traced to a board malfunction in the Black processor and a cabling malfunction between the COMSEC equipment and the Black processor. While there, BBN conducted equipment demonstrations and training sessions on the MAT for shipboard personnel. Considerable inconvenience was involved in this effort because of repeated changes in travel plans.

As a result of this trip, shipboard personnel were able to exchange information with CMU personnel over the ARPANET, despite difficulties in coordinating satellite time and in scheduling NOSC resources. Malfunctions also occurred in the ACCAT TOPS-20 computer when it was accessed by the ship and in the ACCAT Private Line Interface (PLI). Because the SHIP2 Black processor at NOSC was cannibalized for system operations, we had to transfer the TIU of the SHIP2 MAT to the SHORE1 MAT, to allow NOSC to communicate directly with the ship. Since MATNET is a secure network and CMU has only a non-secure connection to the ARPANET, Tracor and NOSC personnel were required to hand retype messages to effect Red-to-Black data transfers; Black-to-Red data transfers were accomplished via floppy disks written on an Apple III personal computer for temporary data storage.

Other activities during the last quarter are as follows. After taking delivery of another C/30 packet switch processor, we

implemented all the modifications necessary to enhance survivability in a seagoing environment (see BBN Combined Quarterly Technical Report No. 27). This unit, which was named Satellite IMP #6 consistent with its inclusion in the SHIP6 site, was tested at BBN and subsequently shipped by air freight to ECI for integration with associated Black equipment and TIU. Consequently, ECI has two complete MAT stations, allowing two-node contention packet testing to be run at that site.

Conversion of MATNET from a Class A net to a Class B net, which was initiated by the Internet Configuration Control Board, requires modifications in the Satellite IMP macrocode, the gateway software, and the TIU software to implement the new 16-bit Class B net number. SRI International, having released a preliminary version of the new TIU software written in C language and accommodating Class B net numbers, expressed uncertainty about the adequacy of performance of the new software with the limited memory space of the MATNET LSI-11/02 TIU hardware. Limited buffering may restrict the maximum number of simultaneous TIU users to two or three. When MATNET interacts directly with the rest of the non-secure internet system, the new numbering system requires MATNET to have implemented Class B net numbers; hence, we intend to investigate the feasibility of using the new TIU software.

We also made extensive revisions to the document describing the Satellite IMP and the Red subsystems, and we gave briefings to the new commanding officer of the CVN70, Captain Tom Mercer, and to the NAVELEX Technology Meeting held in Washington, D.C.

DISTRIBUTION

ARPA
Director (3 copies)
Defense Advanced Research Projects Agency
1400 Wilson Blvd.
Arlington, VA 22209
Attn: Program Manager
R. Kahn
R. Ohlander
D. Adams
B. Leiner

DEFENSE DOCUMENTATION CENTER (12 copies)
Cameron Station
Alexandria, VA 22314

DEFENSE COMMUNICATIONS ENGINEERING CENTER
1860 Wiehle Road
Reston, VA 22090
Attn: Maj. J. Fredricks

DEPARTMENT OF DEFENSE
9800 Savage Road
Ft. Meade, MD 20755
Attn: R. McFarland C132 (2 copies)

DEFENSE COMMUNICATIONS AGENCY
8th and South Courthouse Road
Arlington, VA 22204
Attn: Code B645
Glynn Parker, Code B626

NAVAL ELECTRONIC SYSTEMS COMMAND
Department of the Navy
Washington, DC 20360
Attn: B. Hughes, Code 6111
F. Deckelman, Code 6131

MIT Laboratory for Computer Science
545 Technology Square
Cambridge, MA 02138
Attn: D. Clark

MIT Lincoln Laboratory
244 Woods Street
Lexington, MA 02173
Attn: C. Weinstein

DISTRIBUTION cont'd

BOLT BERANEK AND NEWMAN INC.
1300 North 17th Street
Arlington, VA 22209
Attn: E. Wolf

BOLT BERANEK AND NEWMAN INC.
10 Moulton Street
Cambridge, MA 02238

S. Blumenthal
M. Brescia
R. Bressler
P. Cudhea
W. Edmond
L. Evenchik
G. Falk
J. Goodhue
S. Groff
R. Gurwitz
J. Haverty
F. Heart
J. Herman
R. Hinden
D. Hunt
S. Kent
A. McKenzie
D. McNeill
D. Melone
W. Milliken
M. Nodine
R. Rettberg
J. Robinson
E. Rosen
P. Santos
J. Sax
S. Storch
R. Thomas
B. Woznick
Library